

CAJ/GLM:ach:kaa 10/13/05 439237 95486.2
PATENT

Attorney Reference Number 3382-61916-01
Application Number 10/071,526

Remarks:

Reconsideration of the application is respectfully requested in view of the foregoing amendments and following remarks. Claims 1-15, 17-20, 22-32 are pending in the application. No claims have been allowed. Claims 1, 12, 13, 17, 24, 27 and 28 are independent. Claims 1, 5, 6, 12, 13, 17, 24, 27 and 28 have been amended. Claim 21 has been canceled without disclaimer or prejudice to renewal.

Cited Art

The final Office Action dated June 13, 2005 cites U.S. Patent No. 5,586,304 to Stupek, Jr. et al. ("Stupek") and U.S. Patent No. 6,802,061 to Parthasarathy ("Parthasarathy").

Interview

Applicants thank the Examiner and Primary Examiner Anil Khatri for their time during the telephonic interview on September 20, 2005. Claims 1, 5, 6, 13, 17, and 21 were discussed. Applicants believe the interview was helpful and now present amended claims for further consideration. During the interview, the feature relating to recursion in claim 21 was viewed in a favorable light. Applicants have added language to all independent claims regarding the recursion feature and have also clarified other aspects as recited.

102 Rejection

Patentability of Claims 1-9, 11-15, 17-20, 22-29, and 32 over Stupek under § 102(b)

The Action rejects claims 1-9, 11-15, 17-20, 22-29, and 32 under 35 U.S.C. § 102(b) as being anticipated by Stupek. Applicant respectfully submits that the claims in their present form are allowable over the cited art. To establish a *prima facie* case of anticipation, the cited art must show each and every element as set forth in a claim. MPEP § 2131.01.

CAJ/GLM:ach:kaa 10/13/05 439237 95486.2
PATENT

Attorney Reference Number 3382-61916-01
Application Number 10/071,526

Claim 1

Claim 1 has been amended to clarify “after acquiring the software associated with the software dependency, updating a database at the computer indicating the software associated with the software dependency is installed on the computer ... wherein the acquiring the software associated with the software dependency comprises recursively processing the one or more other software dependencies.” Amended claim 1 reads as follows (emphasis added):

In a computer, a method of processing one or more software dependencies, the method comprising:
for one or more of the software dependencies, determining whether software associated with the software dependency is present on the computer;
responsive to determining the software associated with the software dependency is not present on the computer, acquiring the software associated with the software dependency; and
after acquiring the software associated with the software dependency, updating a database at the computer indicating the software associated with the software dependency is installed on the computer;
wherein at least one of the software dependencies refers to a list comprising one or more other software dependencies; and
wherein the acquiring the software associated with the software dependency comprises recursively processing the one or more other software dependencies.

Stupek’s description of dependencies does not anticipate processing software dependencies by “responsive to determining the software associated with the software dependency is not present on the computer, acquiring the software associated with the software dependency ... wherein the acquiring the software associated with the software dependency comprises recursively processing the one or more other software dependencies” as recited by claim 1. For example, the Application at p. 4, lines 7-16 describes the process of acquiring dependencies as follows:

A software package manager uses a distribution unit containing components for a software package and a manifest file that describes the distribution unit to manage the installation, execution, and uninstallation of software packages on a computer. For installation, the package manager acquires the manifest file and parses it to learn if the software package depends on any additional components.

CAJ/GLM:aeh:kan 10/13/05 439237 95486.2
PATENT

Attorney Reference Number 3382-61916-01
Application Number 10/071,526

The package manager resolves any dependencies by acquiring a distribution unit containing the needed component and installs the dependency's distribution unit as described below. Because dependencies can be nested within dependencies, the package manager recursively processes all the dependencies before finishing the installation of the software package that depends upon the additional components.

Regarding dependencies, Stupek describes that:

While each upgrade distribution medium will commonly contain all upgrade packages upon which a particular upgrade depends, it is also likely that upgrades to a package will depend upon upgraded packages not stored on the distribution medium. Stupek col. 6, lines 35-39.

Stupek further describes that:

Therefore, the dependency information in the Package database 25 describes not only the dependencies between packages on the CD, but also all dependencies between an upgrade package and any upgrade not available on the CD. Even though the unavailable upgrades cannot be automatically installed with the available upgrades, the user is nonetheless aware of their necessity. Stupek col. 6, lines 45-51.

Accordingly, Stupek's shortfall is illustrated by Stupek's example of a word processor upgrade that is dependent upon a printer driver upgrade where the printer driver upgrade (a dependency) cannot be automatically installed if it is not on the distribution medium along with the word processor upgrade. Stupek col. 6, lines 39-45.

Applicants recognized this situation described by Stupek regarding dependencies as a problem. For example, the Application at p. 3, lines 14-20 and Fig. 10 describe the problem with obtaining dependencies as follows:

Although most current software is written in modules, there is no current mechanism that handles the situation where one component in a software program requires the presence of another to operate. If a user downloads software from a Web page, the user may discover that the program requires an external library which necessitates another network session to download, assuming the user can find the right location, and then the user must manually install the library before installing the software.

CAJ/GLM:ach:kaa 10/13/05 439237 95486.2
PATENT

Attorney Reference Number 3382-61916-01
Application Number 10/071,526

Applicants solve this problem by acquiring dependencies. For example, the Application at p. 4, lines 11-20 describes this process as follows:

The package manager resolves any dependencies by acquiring a distribution unit containing the needed component and installs the dependency's distribution unit as described below. Because dependencies can be nested within dependencies, the package manager recursively processes all the dependencies before finishing the installation of the software package that depends upon the additional components.

The software package manager acquires the distribution unit and extracts the components in the distribution unit into a directory on the computer. The package manager causes the operating system of the computer to install the software. The package manager then updates a code store data structure with information in the manifest file.

Because Stupek describes that dependencies that are not on the distribution medium "cannot be automatically installed," Stupek does not anticipate the "responsive to determining the software associated with the software dependency is not present on the computer, acquiring the software associated with the software dependency ... wherein the acquiring the software associated with the software dependency comprises recursively processing the one or more other software dependencies" language of claim 1. Therefore, claim 1 should be in condition for allowance.

Claims 2-9, 11

Claims 5 and 6 have been amended to move language to claim 1. Claims 2-9 and 11 ultimately depend on claim 1. Thus, at least for the reasons set forth above with regard to claim 1, claims 2-9 and 11 are in condition for allowance.

CAJ/GLM:seh:kaa 10/13/05 439237 95486.2
PATENT

Attorney Reference Number 3382-61916-01
Application Number 10/071,526

Claim 32

Claim 32 depends on claim 1. Thus, at least for the reasons set forth above with regard to claim 1, claim 32 is in condition for allowance.

Claim 12

Claim 12 is directed to a computer-readable medium for performing the method of claim 1. Claim 12 has been amended to correspond with claim 1. Therefore, at least for the reasons set forth above with regard to claim 1, claim 12 is in condition for allowance.

The Remaining Claims

Applicants have amended each of the remaining independent claims (13, 17, 24, 27, and 28) to also reflect language regarding recursion, and other language has also been added. Rather than belabor the language of each of these claims, Applicants point out that each claim recites a novel and non-obvious combination allowable over the cited art.

Similarly, the respective dependent claims, 14-15, 18-20, 22-23, 25-26, and 29-31 are allowable.

103 Rejection

Patentability of Claims 10, 30, and 31 over Stupek in view of Parthasarathy under § 103(a)

The Action rejected claims 10, 30, and 31 under 35 U.S.C. § 103(a) as unpatentable over Stupek in view of Parthasarathy. Applicants respectfully submit the claims in their present form are allowable over the cited art.

To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the

CA/GLM:ueh:ksa 10/13/05 439237 95486.2
PATENT

Attorney Reference Number 3382-61916-01
Application Number 10/071,526

claim limitations. MPEP § 2142. Motivations to combine or modify references must come from the references themselves or be within the body of knowledge in the art. MPEP § 2143.01.

Claims 10, 30, and 31

Claim 10 ultimately depends from claim 1, and claims 30 and 31 depend from claim 28. Therefore, for at least the reasons stated above with regard to Stupek for claims 1 and 28, the Action's § 103(a) rejection of claims 10, 30, and 31 also cannot be supported. Thus, the claims are in condition for allowance.

Request for Interview

If any issues remain, the Examiner is formally requested to contact the undersigned attorney prior to issuance of the next Office Action in order to arrange a telephonic interview. It is believed that a brief discussion of the merits of the present application may expedite prosecution. Applicants submit the foregoing formal Amendment so that the Examiner may fully evaluate Applicants' position, thereby enabling the interview to be more focused.

This request is being submitted under MPEP § 713.01, which indicates that an interview may be arranged in advance by a written request.

CAJ/GLM:neh:kna 10/13/05 439237 95486.2
PATENT

Attorney Reference Number 3382-61916-01
Application Number 10/071,526

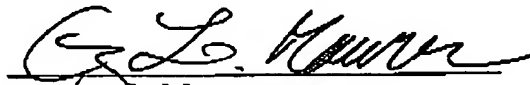
Conclusion

The claims in their present form should now be allowable. Such action is respectfully requested.

Respectfully submitted,

KLARQUIST SPARKMAN, LLP

By


Gregory L. Maurer
Registration No. 43,781

One World Trade Center, Suite 1600
121 S.W. Salmon Street
Portland, Oregon 97204
Telephone: (503) 226-7391
Facsimile: (503) 228-9446